LV1b Downlink Test #3

07-16-2000

Summary

We finally completed the critical "does it actually work" test of the 900MHz LV1b Downlink system. The system worked well enough in these 12 tests over a range of 0.85 miles that we are fairly confidant the system will work for LV1b. It will require more tests, however, for us to be very confident in the whole LV1b RF system (see the "Future Tests" below).

Our Plan

Our plan was to get the 900MHz system up and running locally, and then try a long distance "near apogee" test (12,000ft or 2.3 miles). A computer hooked to the transmitter would then send computer files over the 900MHz downlink that would then be captured by a computer hooked to the receiver.

The Reality

Of course, most of the time for the test (10:00am - 2:30pm) was spent locally trying to get the bloody thing to work, and then driving out to the testing location. Thankfully, the testing went very well.

We removed the transmitter from the payload and attached it to a plywood board, along with some power supplies and wires. This turned out to be a great thing, as it was easy to transport and debug the transmit module. The receiver was already boxed up, so we left it in the box on the birdfeeder antenna and went from there.

We initially ran into a very strange problem where the transmitter would only key up if the serial input line was held high; that problem mysteriously cleared up when we took the transmitter into the basement for further testing. After it suddenly began working, we went across the field from Andrew's house (around 300ft) and it worked perfectly.

So we were off to the distance test. We came up with the idea of trying to go between the Crown Point Outlook in the Columbia Gorge (that cool round building sticking out on the point) and small outlook called the "Portland Women's Forum" Outlook on the same road a few miles away. Nathan and Andrew took the transmitter and nosecone in Andrew's car and went to the PWF Outlook and Glenn and Dennis took the receiver in Glenn's car and went to Crown Point. We ran tests from 2:30pm until about 4:15pm and everything seemed to go ok besides a few minor glitches. Bart and family even showed up for a awhile after helping us out at Andrew's house.

We did twelve tests altogether; see the testing section below for details. Afterward, we drove back to Portland and just for giggles did a transmit/receive test from Glenn's house to Andrew's house (about ½ mile away). Despite the trees in the way and lack of line of site path, we still got data through (although close to the noise floor at 90dBm). Fun!

Later, we found out (via mapquest.com's topo map service) that the ~1 to ~1.5 mile distance between the two outlooks was only 0.85 miles. Ugh! That's disappointingly close together (that's only 1/3 of the apogee height). Despite the closeness, we still think the test was a success since a factor of 3 may not make as much of a difference as it might seem. Worst case scenario is that we use the YAGI antenna with a gain of +9dB to capture the signal.

Future Tests (LV1b Downlink Test #4)

We really need to go at least 2.5 miles for the next test. Glenn and Andrew identified two potential sites: one is from Crown Point to the Washington side Columbia River beach, off of the nature reserve on the Columbia, the other is again from Crown Point to Biddle Butte (Mt. Zion), on a small road called Mt. Zion Road. Directions: Take Highway 14 east to Belle Center Rd north, then Strunk Rd east (it may start as Mt. Pleasant Rd and then branch right to Strunk Rd) and then finally go south on Mt. Zion Rd.

The next test must involve all RF systems: The 75cm ATV system, the 2m DTMF uplink radio, and of course the 900MHz downlink system. Testing all of these systems together, at an apogee-like distance, will give us a high degree of confidence about how the flight will go.

Thoughts for the next test:

- We need a tripod system for holding and pointing antennas.
- Bring the digital camera.
- Don't use inverters (use onboard batteries) for the next test so we're as close to flight mode as possible.

- Don't forget to get the actual signal strength in dB.
- Try and put everything on one board (like Test #3) - it greatly simplifies the task.
- The FCSView will need to store the packets and it would be very useful to count how many packets are good and how many are bad (Checksum?)
- Bring the GPS for distance measurement.
- Bring Binoculars if possible.
- Can we get better serial port software than Kermit or Commhex that isn't dependent on ASCII data?

Test #1	
Transmitter:	Nosecone pointed up, antennas broadside to receiver
Receiver:	Vertically polarized YAGI pointing at transmitter
Length:	40 seconds
Filename:	F1.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	Our first test. We found out that Glenn's 2m radio screws up reception when he was transmitting (even when his radio was in low power mode). And of course, low power mode (power amp off) on the transmitter didn't work. Finally, we got a successful file transfer. Some weird binary characters.

Test #2	
Transmitter:	Nosecone pointed up, antennas broadside to receiver (same as #1)
Receiver:	Vertically polarized YAGI pointing at transmitter (same as #1)
Length:	40 seconds
Filename:	F2.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	Repeat of the first test in case binary characters was due to Kermit.

Test #3	
Transmitter:	Nosecone pointed up, antennas towards receiver (90° from broadside)
Receiver:	Vertically polarized YAGI pointing at transmitter (same as #1)
Length:	40 seconds
Filename:	F3.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	We decided to turn the transmitter to low power mode between tests #2 and #4 and turn it on right before transmitting. Because of this, the receiver received a bunch of garbage at the beginning of F3.log.

Test #4a	
Transmitter:	Nosecone pointed up, antennas broadside to receiver (same as #1)
Receiver:	Birdfeeder antenna pointing up
Length:	--
Filename:	--
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	This test didn't go well. We kept receiving garbage, which stopped Kermit from operating properly. We cancelled this test and decided to try the YAGI again.

Test #4b	
Transmitter:	Nosecone pointed directly away from receiver
Receiver:	Vertically polarized YAGI pointing at transmitter (same as #1)
Length:	40 seconds
Filename:	F4.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	Some garbage at the end of F4.log. It turns out it doesn't matter which direction the YAGI is pointed to get a good

signal! Neat.

Test #5	
Transmitter:	Nosecone pointed directly at receiver
Receiver:	Vertically polarized YAGI pointing at transmitter
Length:	40 seconds
Filename:	F5.log
Serial Prog.:	Kermit
Test Type:	Printable character test (All printable ASCII characters)
Notes:	Ok.

Test #6	
Transmitter:	Nosecone pointed directly at receiver (same as #5)
Receiver:	Vertically polarized YAGI pointing at transmitter (same as #5)
Length:	40 seconds
Filename:	F6.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	CommHex
Notes:	Tried CommHex instead of Kermit as serial I/O program since Kermit kept having problems with garbage.

Test #7	
Transmitter:	Nosecone pointed up, antennas broadside to receiver (same as #1)
Receiver:	Vertically polarized YAGI pointing at transmitter (same as #1)
Length:	40 seconds
Filename:	F7.log
Test Type:	Binary tests (0x00 - 0xFF)
Serial Prog.:	CommHex
Notes:	First binary test.

Test #8	
Transmitter:	Horizontally polarized nosecone pointed directly away from receiver
Receiver:	Bird feeder antenna pointed towards transmitter
Length:	40 seconds
Filename:	F8.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	Kermit
Notes:	First test of the "Apogee Configuration" - rocket pointed up (away from receiver), birdfeeder antenna pointed towards rocket. This should be what it's like when the rocket is near or at apogee.

Test #9	
Transmitter:	Horizontally polarized nosecone pointed directly away from receiver (same as #8)
Receiver:	Bird feeder antenna pointed towards transmitter (same as #8)
Length:	40 seconds
Filename:	F9.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	CommHex
Notes:	Same as Test #8, but we switched to CommHex.

Test #10	
Transmitter:	Vertically polarized nosecone pointed directly away from receiver (same as #8 but vertically polarized)
Receiver:	Bird feeder antenna pointed towards transmitter (same as #8)
Length:	40 seconds
Filename:	F10.log

Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	CommHex
Notes:	Tested if polarization of nosecone affects transmission.

Test #11	
Transmitter:	Horizontally polarized nosecone with antenna tip null pointed directly at receiver (close to #8)
Receiver:	Bird feeder antenna pointed towards transmitter (same as #8)
Length:	40 seconds
Filename:	F11.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	CommHex
Notes:	Same as Test #8, but we switched to CommHex.

Test #12	
Transmitter:	Nosecone pointed directly away from receiver (same as #8)
Receiver:	Bird feeder antenna pointed towards transmitter (same as #8)
Length:	5 minutes
Filename:	F12.log
Test Type:	Printable character test (All printable ASCII characters)
Serial Prog.:	CommHex
Notes:	Five minute test. Last test.